The 10th INTERNATIONAL DAAAM SYMPOSIUM
Vienna University of Technology, Vienna, Austria
21-23rd October 1999

SUPPORT FOR PROBLEM FORMALISATION IN ENGINEERING DESIGN: AN ENQUIRY INTO THE ROLE OF KNOWLEDGE-LEVEL MODELS

Džbor, M.

Abstract: *Engineering design often deals with ill-defined problems. For such problems it is typical that they arise from situations where some goal is desired but the way to achieve it is unclear. However, designers may know how to achieve the goal; they might have acquired experience from similar situations in the past. Later they may draw upon that experience.*

Engineering design (ED) contains two basic activities – problem description and solution. Historically ED was more concerned with the transition from the description to the solution than with the issue of how faithfully the description really addressed the problem at hand. Finding the relevant description of a problem is as important as finding a solution to it. Thus far, there has been little systematic study of how designers can be supported in the initial phases of design.

This paper reviews current understanding of how designers tackle problems in uncertain situations, defines problem formalisation, and summarises features necessary for the support.

Key words: *engineering design, problem formalisation, intelligent support, knowledge-level models, ontologies.*

1. INTRODUCTION

Simon (Simon, 1973) identified features typically violated by ill-defined problems. The characteristics include the availability of a well-defined problem space for problem representation, and the existence of a criterion to test any problem description and/or solution. ED is defined as a transformation of initial requirements into description of a system that meets them. And requirements are often vague and unclear, i.e. ill-defined.

Schön points out that in practice designers are not presented with a detailed description of problems. The descriptions must be built from the uncertain situations (Schön, 1983). Designers have to perform a certain amount of work to convert situations in soluble problems. Dominowski and Dallob see problems as situations in which some goal is desired but the way to achieve it is unclear (Dominowski & Dallob, 1995). However, designers might have acquired knowledge how a problem might be tackled. *Problem formalisation* is defined as a transformation of an uncertain situation to a set of soluble problems described in a domain specific language. It contains basically two actions: problem *recognition* and problem *description*.

Designers involved in ED often draw on their previous experience. Schön's studies of designers reveal as a key feature designers' ability to see similarity between the current and pre–vious problems; though they need not to be aware of specific corresponding attributes. In other words, designers discover similarity not on the structural level of problems, but a more abstract level; they describe design goals or desired functionality.

Though problem formalisation is a highly creative activity dependent on the subject involved, the knowledge-level models might support it. A knowledge-level model (KlM) is a representation of knowledge that is relevant to a particular subject. KlM clarify the structure of designers' knowledge about the current situation; they provide a common vocabulary to describe situations in a particular domain consistent with designers' empirical and theoretical knowledge. They can also support reflection on the current problem description, and an insightful approach.

2. NATURE OF PROBLEMS IN DESIGN

Schön sees problem formalisation (PF) as *a process, where designers interactively name objects, TO which they will attend and frame the context, IN which they will attend to the objects* (Schön, 1983). His definition supports the claim that problem formalisation contains both problem recognition and description. Problem recognition builds a framework for viewing an uncertain situation. Once the viewpoint is chosen, designers name concepts that are important and relevant from the current point of view on the situation.

Problem formalisation is an iterative process. It means that one can change the current perspective if it does not perform satisfactorily. The change will cause new concepts to be identified within the situation, which may further lead to the possibility of applying another perspective to describe the current situation. The need to change a perspective may be caused by an unexpected result in the current perception of a problem. Schön refers to it as 'a surprise' and claims that any result inconsistent with designers' knowledge might be perceived as 'a surprise'.

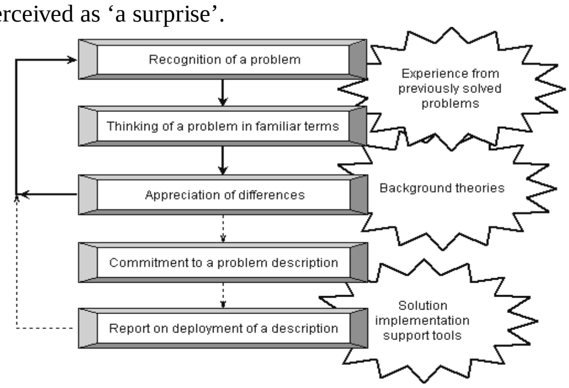

Fig. 1. Problem formalisation and reflection

When designers find something surprising in the problem description or the chosen perspective, reflection occurs. First, designers have recognised and described a problem using their empirical and theoretical knowledge. Knowledge re-use has been performed through thinking about the problem in terms of a familiar situation. Familiar situations are retrieved from a design repository; differences between the previous and current situation might be perceived. Positive or negative differences may be expected, and designers can reflect on their perspectives and/or descriptions of a problem. Reflection is depicted as loop in Fig. 1. An uncertain situation is thus tackled iteratively; problem is continually re-framed and re-described until a coherent description of situation is attained.

3. CREATIVITY IN ENGINEERING DESIGN

Cognitive scientists warn that designers can fail to solve problems due to fixation on familiar interpretation of the situation (Dominowski & Dallob, 1995). The fixation might be avoided by an insightful approach; i.e. a sudden understanding of a problem, especially by intuition. Cross suggests an idea of 'creative leaps' based on the observation that many innovative design concepts emerge suddenly, in a form of 'enlightenment'. He defines creative leap as *a sudden perception of a completely new perspective on the situation* (Cross, 1997). In practice it might be also a shift made in an existing perspective.

Cross observed bike rack designers (Cross, 1997), and reports that the initial problem recognition and designers' experience guided their work. When they attempted to solve the problem as they specified it in the beginning, they soon recognised new issues that had to be included in the problem description. They proposed an approach to tackle the situation early in the design process, but all subsequent refinements of it did not solve a principal issue (mud spray thrown by a rear wheel). They reflected on the problem description, and the breakpoint came when they looked at rack as a tray, or '*something like a bag*'. This newly identified concept caused a shift in their perspective. They have taken the familiar concept of 'bag' and explored various modifications arising from it. A familiar situation can give rise to creative design. Cross has identified several models of creative design. The most common are: *combination*, *mutation* and *analogy*. From them, the first two can be supported in a limited way. KlM-based approach presented in this paper comes out of the third design model – analogy.

Similarity and insight are combined in a claim that a key factor in the creative design is designer's *insight by which certain focuses are chosen, and combined in an innovative way* (Candy & Edmonds, 1996). Candy & Edmonds define design as a unique case extending current designers' knowledge. In their study it was designer's *ability to make analogies with products in other areas* that extended the traditional (bicycle) design space. Candy & Edmonds agree that problem restructuring can create a potential for insight, and thus creative design.

4. KNOWLEDGE-LEVEL MODELS IN DESIGN

Knowledge-level models (KlM) date back to Newell's claim that an agent's knowledge about a problem significantly influences its solution. One approach to the construction of KlM is based on common ontologies. Gruber (Gruber, 1993) defines an ontology as *an explicit specification of conceptualisation; in AI what 'exists' is that what can be 'represented'*. Chandrasekaran sees an ontology as a representation vocabulary, specialised to some domain or subject matter (Chandrasekaran *et al.*, 1998). Ontologies clarify the structure of knowledge in a particular domain, define basic concepts and relations; and support knowledge re-use. The structuring role of ontologies is emphasised; knowledge re-use is a consequence of a clear structure.

For a problem domain several ontologies may be developed that will differ in their specificity and viewing perspective. According to the first criterion 'upper' and 'lower' ontologies are distinguished. Upper ontologies are more general, usually applicable across several domains. They serve as a base for the construction of specific, domain/application oriented ontologies. Approach to design support as proposed in this paper, assumes that designers describe a problem rather in 'upper' and middle-level ontologies in the beginning. Later when the problem is better understood 'lower' ontologies are applied. These may exist in a repository (built in the past), or may be created on top of existing ontologies. A lower ontology from one case can serve as an upper or middle-level in the next one. Let us assume that the previous problems were described using KlM and common ontologies. The role of KlM is thus threefold:

- to describe familiar ontologies to build on top of;
- to serve as 'reference ontologies' of familiar terms for the initial attempts to recognise and describe the problem;
- to provide support in the form of consistency & completeness checking, visualisation of reasoning process, etc.

5. SUPPORT METHODOLOGY, CONCLUSIONS

Based on the given definition of design that it is a goal-oriented process of searching for how to achieve desired functionality, and reviewed cases, the following support methodology is suggested: two problems can be considered analogous if they have a similar goal. When a goal is fixed, and KlM exist for the previous problems and domain theories, support tool will provide designers with help in form of which functions, behaviours and/ or components are relevant to be included in current description of a problem. The procedure is depicted in Fig. 2.

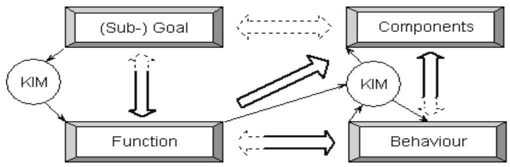

Fig. 2. Problem formalisation process incl. support

It can be illustrated by a simple example of dynamic system simulation, where the designer has experience e.g. from pendulum simulation, and a task to simulate an electric circuit. S/he describes the goal as a simulation of dynamics, and identifies the domain as electro-magnetic. By analogy with pendulum concepts such as *disturbance*, *equilibrium*, and *source of disturbance* are found. Now s/he may re-frame the problem to look for sources that can cause disturbance in electric circuit. Using domain theory behavioural concepts as *inductance* and *resistance* are named, and cause re-framing again. The designer may define components and their relations to perform that particular behaviour and functionality. Another shift of perspective can bring a concept of *conductance*; current description can be refined until the designer is satisfied with his/her model.

Support is provided in the form of concepts that exist in KlM for a previous problem or a domain theory. It is on designers to accept concepts in the KlM for the current problem, or refuse them. Level of support will depend on designers' experience; novices may benefit more than experts do. This must be proved by experiments with a prototype tool being developed.

Theoretical and empirical knowledge, current and previous problems can be described using a vocabulary of terms from the domain ontology that allows the problem descriptions to be easily compared, and re-used. Providing designers with support for the problem formalisation from the very beginning can result in saving their efforts in this area, which they can invest in their creativity. This will be also studied in future experiments.

6. ACKNOWLEDGEMENTS

I am grateful to my colleagues and PhD advisers at KMi who revised early versions of this paper. It would not be possible without help from *Enrich* – EU funded project, No. 29015.

7. REFERENCES

Candy, L. & Edmonds, E. (1996). Creative design of the Lotus bicycle: implications for knowledge support systems research. *Design Studies,* **17,** pp. 71-90.
Chandrasekaran, B., Josephson, J.R. & Benjamins, V.R. (1998) The Ontology of Tasks and Methods. In *11th Banff Knowledge Acquisition for KBS Workshop*, Vol. 1, Canada.
Cross, N. (1997). Descriptive models of creative design: application to an example. *Design Studies,* **18,** pp. 427-440.
Dominowski, R.L. & Dallob, P. (1995). Insight and Problem Solving. In *The Nature of Insight* (Eds. Sternberg, R.J. & Davidson, J.E.) MIT Press, USA, pp. 33-62.
Gruber, T.R. (1993). A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition,* **5**.
Schön, D.A. (1983) Reflective Practitioner – How professionals think in action, Basic Books, Inc., USA.
Simon, H.A. (1973). The structure of ill-structured problems. *Artificial Intelligence,* **4,** pp. 181-201.

Author: *Martin* ***DŽBOR***, KMi, The Open University, Milton Keynes, MK7 6AA, UK; Fax: +44 (0)1908 653169; E-mail: M.Dzbor@open.ac.uk; http://kmi.open.ac.uk/people/dzbor